

The diagram illustrates a computer system architecture with the following components and connections:

- 300 Instruction Memory:** Connected to the Instruction Control Unit (101) via a bidirectional bus (600).
- 100 System Bus:** Connects the Instruction Memory (300) to the Instruction Control Unit (101).
- 101 Instruction Control Unit:** The central control unit, connected to the Integer Execution Unit (102) and Load/Store Unit (103) via bidirectional buses (102 and 103).
- 102 Integer Execution Unit:** Connected to the Load/Store Unit (103) via a bidirectional bus (104).
- 103 Load/Store Unit:** Connected to the Integer Execution Unit (102) via a bidirectional bus (105).
- 107 Context Switch Control Unit:** Connected to the Integer Execution Unit (102) via a bidirectional bus (106).
- 200 Data Memory:** Connected to the Integer Execution Unit (102) and Load/Store Unit (103) via a bidirectional bus (800).
- Peripheral Units:** Connected to the system via a bidirectional bus (400).

The diagram also shows internal data paths within the Integer Execution Unit (102) and Load/Store Unit (103), including registers and ALUs.

(19) 日本国特許庁 (J P)

(12) 公表特許公報 (A)

(11) 特許出願公表番号

特表2002-513182

(P2002-513182A)

(43) 公表日 平成14年5月8日(2002.5.8)

(51) Int.Cl. ⁷	識別記号	F I	テマコード [*] (参考)
G 0 6 F 9/46	3 1 3	G 0 6 F 9/46	3 1 3 C 5 B 0 3 3
9/42	3 3 0	9/42	3 3 0 R 5 B 0 9 8

審査請求 未請求 予備審査請求 有 (全 42 頁)

(21) 出願番号 特願2000-546304(P2000-546304)
 (86) (22) 出願日 平成11年2月25日(1999.2.25)
 (85) 翻訳文提出日 平成12年10月26日(2000.10.26)
 (86) 国際出願番号 PCT/US99/04134
 (87) 国際公開番号 WO99/56209
 (87) 国際公開日 平成11年11月4日(1999.11.4)
 (31) 優先権主張番号 09/069,030
 (32) 優先日 平成10年4月27日(1998.4.27)
 (33) 優先権主張国 米国 (US)
 (81) 指定国 EP(AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, I T, LU, MC, NL, PT, SE), JP, KR

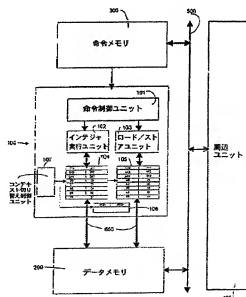
(71) 出願人 インフィニオン テクノロジーズ ノース
 アメリカ コーポレーション
 Infineon Technolog
 ies North America Co
 rp
 アメリカ合衆国 カリフォルニア サン
 ホセ ノース ファースト ストリート
 1730
 (72) 発明者 ロジャー ディー アーノルド
 アメリカ合衆国 カリフォルニア サニー
 ヴェール キャリック コート 566
 (74) 代理人 弁理士 矢野 敏雄 (外3名)

最終頁に続く

(54) 【発明の名称】 コンテキスト切り替え能力を有する装置

(57) 【要約】

本発明は、少なくとも1つの読み取りポートと1つの書き込みポートを有する少なくとも1つのレジスタを含む、データ処理ユニットに関する。このレジスタは少なくとも2つのメモリスセルを有しており、メモリスセルのそれぞれは1つの書き込みラインと1つの読み取りラインを有している。さらに前記レジスタは、前記メモリスセルの1つの前記読み取りラインを前記読み取りポートと結び付けるための、複数のインプットと1つのアウトプットを有する第1切り替え装置と、前記メモリスセルの1つの前記書き込みラインを前記書き込みポートに結び付けるための、第2切り替え装置とを含んでいる。



【特許請求の範囲】

【請求項1】 データ処理ユニットにおいて、

少なくとも1つの読み取りポートおよび1つの書き込みポートを有する少なくとも1つのレジスタを含み、

前記レジスタは少なくとも2つのメモリセルを有し、

前記メモリセルのそれぞれは、1本の書き込みラインおよび1本の読み取りラインを有し、

第1のスイッチを含み、

前記第1スイッチは、前記メモリセルの1つの前記読み取りラインを前記読み取りポートに結合させるための、複数のインプットおよび1つのアウトプットを含み、

第2のスイッチを含み、

前記第2のスイッチは、前記メモリセルの1つの前記書き込みラインを前記書き込みポートに結合させるための、少なくとも1つのインプットおよび複数のアウトプットを有する、

ことを特徴とするデータ処理ユニット。

【請求項2】 さらに1つのインプットと1つのアウトプットを有する多数の読み取りポートドライバを含み、

複数の読み取りポートデータラインを含み、

それによって前記読み取りポートドライバの前記インプットが前記スイッチの前記アウトプットに接続され、

前記読み取りポートドライバの前記アウトプットが前記読み取りポートデータラインに接続されている、請求項1に記載のデータ処理ユニット。

【請求項3】 さらに複数の書き込みポートデータラインを含み、

前記第2のスイッチが、それぞれの書き込みポートデータラインを、それぞれのメモリセルの前記書き込みラインにつなぐことができる、請求項1に記載のデータ処理ユニット。

【請求項4】 前記第2のスイッチが、それぞれのメモリセルへのランダムアクセスを提供する、請求項1に記載のデータ処理ユニット。

【請求項5】 前記第2のスイッチが、それぞれのメモリセルのそれぞれの書込みラインを、前記書込みポートの1つと結合させるための複数の転送レジスタを含む、請求項4に記載のデータ処理ユニット。

【請求項6】 前記レジスタの内容を蓄積し、そしてロードするための制御ユニットをさらに含む、請求項1に記載のデータ処理ユニット。

【請求項7】 前記制御ユニットが状態マシンを含む、請求項6に記載のデータ処理ユニット。

【請求項8】 データ処理ユニットにおいて、
レジスタの第1のセットとレジスタの第2のセットとを有する、1つのレジスタファイルを含み、

レジスタの前記第1セットにおける前記レジスタが、少なくとも1つの読み取りポートと1つの書込みポートを有し、

前記レジスタのそれぞれは少なくとも2つのメモリセルを有し、

前記メモリセルのそれぞれは1つの書込みラインおよび1つの読み取りラインを有し、

前記第1のレジスタセットのそれぞれのレジスタの前記メモリセルの1つの前記読み取りラインを、それぞれの読み取りポートに結合させるための第1のスイッチを含み、

前記第1のレジスタセットのそれぞれのレジスタの前記メモリセルの1つのそれぞれの書込みラインを、前記書込みポートに結合させるための、第2のスイッチを含む、

ことを特徴とするデータ処理ユニット。

【請求項9】 前記レジスタのそれぞれが、多数の読み取りおよび多数の書込みポートを含み、

前記第1および前記第2のスイッチは、前記多数の読み取りおよび書込みポートを、前記第1のレジスタセットのそれぞれのレジスタのそれぞれのメモリセルに結合させる、請求項8に記載のデータ処理ユニット。

【請求項10】 第1のレジスタセットの前記レジスタの内容を蓄積し、そしてロードするための制御ユニットをさらに含む、請求項8に記載のデータ処理

ユニット。

【請求項 1 1】 前記制御ユニットが状態マシンを含む、請求項 8 に記載のデータ処理ユニット。

【請求項 1 2】 レジスタの前記第 1 セットが、データおよびアドレスレジスタを含む、請求項 8 に記載のデータ処理ユニット。

【請求項 1 3】 1 つのメモリと複数のレジスタとを有するデータ処理ユニット内での、第 1 のタスクから第 2 のタスクへのコンテキスト切り替えの方法において、

前記複数のレジスタは、前記メモリに結合されて、コンテキストを表しているレジスタのセットを構成し、

前記レジスタのそれぞれは少なくとも 2 つのメモリセルと、前記メモリセル間を切り替える 1 つのセレクトとを有し、

前記コンテキスト切り替え方法が、

第 1 の前もって決定されたイベントの実行によって、前記レジスタのそれぞれにおいて次の利用可能なメモリセルへの切り替えのステップと、

第 2 の前もって決定されたイベントの実行によって、前記レジスタのそれぞれにおいて以前のメモリセルに切り替え戻しするステップと、を含むことを特徴とするコンテキスト切り替えの方法。

【請求項 1 4】 $n-1$ の第 1 のイベントの実行の後の（ここで n はレジスタ内のメモリセルの数に等しい）さらなる第 1 のイベントの実行によって、もう 1 つのメモリセルに切り替えた後に、

前記メモリ内のレジスタの前記セットのすべてのレジスタの前記メモリセルの内容を蓄積するステップを含む、請求項 1 3 に記載の方法。

【請求項 1 5】 前記第 1 の前もって決定されたイベントの実行によって、もう 1 つのメモリセルに切り替わる前に、

前記メモリ内のレジスタの前記セットのレジスタの、前もって定められた数の現行のメモリセルの内容を蓄積するステップを含む、請求項 1 3 に記載の方法。

【請求項 1 6】 前記第 1 の前もって決定されたイベントの実行によって、もう 1 つのメモリセルに切り替わった後で、

前記メモリ内のレジスタの前記セットのレジスタの残っている数の現行のメモリセルの内容を蓄積するステップを含む、請求項15に記載の方法。

【請求項17】 n の第1のイベントの実行の後に、(n は1つのレジスタ内のメモリセルの数に等しい)

さらなる第1のイベントの実行によって、

もう1つのメモリに切り替えた後に、

前記メモリ内のレジスタの前記セットのすべてのレジスタの前記メモリセルの内容を蓄積するステップを含む、請求項14に記載の方法。

【請求項18】 $n-1$ の第2のイベントの実行の後に、(n は1つのレジスタ内のメモリセルの数に等しい)

さらなる第2のイベントの実行によって、

もう1つのメモリセルに切り替えた後に、

前記メモリからのレジスタの前記セットのすべてのレジスタの蓄積されているメモリセルの内容を、前記レジスタ内にロードするステップを含む、請求項14に記載の方法。

【請求項19】 前記第2の前もって決定されたイベントの実行によって、もう1つのメモリセルに切り替わる前に、

前記メモリからのレジスタの前記セットの前もって定められた数のレジスタの蓄積されたメモリセルの内容を前記レジスタ内にロードするステップを含む、請求項15に記載の方法。

【請求項20】 前記第2の前もって決定されたイベントの実行によって、もう1つのメモリセルに切り替わった後で、

レジスタの前記セットの残っている数のレジスタの蓄積されているメモリセルの内容を、前記メモリから前記レジスタへロードするステップを含む、請求項16に記載の方法。

【請求項21】 n の第2のイベントの実行の後に、(n は1つのレジスタのメモリセルの数に等しい)

さらなる第2のイベントの実行によって、

もう1つのメモリセルに切り替えた後に、

レジスタの前記セットのすべてのレジスタの蓄積されているメモリセルの内容を、前記メモリから前記レジスタにロードするステップを含む、請求項17に記載の方法。

【請求項22】 1つのメモリおよび複数のレジスタを有するデータ処理ユニットにおける第1のタスクから第2のタスクへのコンテキスト切り替えの方法において、

前記複数のレジスタは、前記メモリと結合されて、コンテキストを表すレジスタのセットを構成し、

前記レジスタのそれぞれは、少なくとも2つのメモリセルを有し、

それぞれのメモリセルは、少なくとも1つの読み取りポートと1つの書き込みポートと、前記読み取りポートの1つを選択する1つのスイッチとを有し、

前記スイッチは、前記書き込みポートのランダムアクセスのために備えられ、

前記コンテキスト切り替えの方法が、

第1の前もって決定されたイベントの実行によって、前記レジスタのそれぞれにおいて次の利用可能なメモリセルの読み取りポートに切り替わるステップと、

第2の前もって決定されたイベントの実行によって、前記レジスタのそれぞれにおいて以前のメモリセルの読み取りポートに切り替え戻しするステップとを含むことを特徴とする、コンテキスト切り替えの方法。

【請求項23】 $n-1$ の第1のイベントの実行の後に、(n は1つのレジスタ内のメモリセル数に等しい)

さらなる第1のイベントの実行によって、

もう1つのメモリセルの読み取りポートに切り替わった後で、

前記メモリ内のレジスタの前記セットのすべてのレジスタの現行のメモリセルの内容を蓄積するステップを含む、請求項22に記載の方法。

【請求項24】 前記第1の前もって決定されたイベントの実行によって、

もう1つのメモリセルの読み取りポートに切り替える前に、

前記メモリ内のレジスタの前記セットの前もって定められた数のレジスタの現行のメモリセルの内容を蓄積するステップを含む、請求項22に記載の方法。

【請求項25】 前記第1の前もって決定されたイベントの実行によって、

もう1つのメモリセルの読み取りポートに切り替えた後で、

前記メモリ内のレジスタの前記セットの残りの数のレジスタの現行のメモリセルの内容を蓄積するステップを含む、請求項22に記載の方法。

【請求項26】 n の第1のイベントの実行の後に、(n は1つのレジスタ内のメモリセルの数に等しい)

さらなる第1のイベントの実行によって、

もう1つのメモリの読み取りポートに切り替わった後で、

前記メモリ内のレジスタの前記セットのすべてのレジスタの現行のメモリセルの内容を蓄積するステップを含む、請求項22に記載の方法。

【請求項27】 $n-1$ の第2のイベントの実行の後に、(n は1つのレジスタ内のメモリセルの数に等しい)

さらなる第2のイベントの実行によって、

レジスタの前記セットのすべてのレジスタの蓄積されているメモリセルの内容を、前記メモリから前記書き込みポートを通して前記レジスタの中にロードするステップを含む、請求項23に記載の方法。

【請求項28】 前記第2の前もって決定されたイベントの実行によって、前記レジスタのセットの前もって定められた数のレジスタの蓄積されているメモリセルの内容を、前記メモリから前記書き込みポートを通して前記レジスタ内にロードするステップを含む、請求項24に記載の方法。

【請求項29】 前記第2の前もって決定されたイベントの実行によって、前記レジスタのセットの残りの数のレジスタの蓄積されているメモリセルの内容を前記メモリから前記書き込みポートを通して前記レジスタにロードするステップを含む、請求項25に記載の方法。

【請求項30】 n の第2のイベントの実行の後に、(n は1つのレジスタ内のメモリセルの数に等しい)

さらなる第2のイベントの実行によって、

前記レジスタのセットのすべてのレジスタの蓄積されているメモリセルの内容を、前記メモリから前記書き込みポートを通して前記レジスタにロードするステップを含む、請求項26に記載の方法。

【請求項31】 1つのレジスタが、どのレジスタがレジスタのセットに割り当てられているかを示す、請求項22に記載の方法。

【発明の詳細な説明】

【0001】

発明の背景

本発明はコンテキスト切り替え能力を有する装置に関する。ほとんどの埋め込み形の、そして実時間のコントロールシステムは、割込みハンドラおよびソフトウェアによって管理されたタスクがそれぞれ、それら自身の仮想のマイクロコントローラ上で実行されるよう、考慮されているモデルに従って設計されている。そのモデルは一般に実時間実行あるいはオペレーティング・システムのサービスによってサポートされている。それらは基礎をなしているマシンアーキテクチャの特徴および能力のトップ上に層をなしている。仮想のマイクロコントローラは、それ自身の汎用レジスタと、そしてプログラムカウンタのような関連する特殊機能レジスタと、プログラムステータスワードなどを有する1つのタスクとして見ることができる。そしてそれらはそのタスクのコンテキストを表している。公知のシステムの大部分における、これらの仮想のマイクロコントローラの取り扱い、それぞれのコンテキストをセーブして、そして復元するソフトウェアの手段によって実行される。そのために、このようなデータ処理ユニットのためのソフトウェアは、増加する一方の量のメモリを必要とする。そしてコンテキスト切り替えオペレーションのための実行オーバーヘッドはアプリケーションタスクに利用可能な処理帯域幅を減らす。

一般に、コンテキストスイッチはレジスタファイルの少なくともある特定のレジスタの内容が、前もって定められたメモリエリア内に蓄積されて、そしてもう1つのメモリエリアの内容で置き換えられることを必要とする。それによって、それぞれのメモリエリアが、特定のタスクあるいは割り込みサービスルーチンと関係するすべての必要なデータを含んでいる特定のコンテキストを表す。コンテキスト切り替えと呼ばれるこのオペレーションは、いわゆる実時間オペレーティング・システムソフトウェアの中で決定的に時間を消費する部分である。このようなオペレーティング・システムはしばしば、外部か、あるいは内部のイベント上で可能な限り速く反応しなくてはならない、という目的を有する。

【0002】

発明の要約

速いハードウェアで支援されるコンテキスト切り替え能力を有する方法およびデータ処理ユニットを提供することによって、コンテキスト切り替えオペレーションのより速い実行を実現することが、本発明の目的である。この目的は、少なくとも1つのレジスタを含む1つのデータ処理ユニットによって達成される。このレジスタは、少なくとも1つの読み取りポートと、そして1つの書込みポートを有する。このレジスタは、少なくとも2つのメモリセルを有し、それぞれのセルは1つの書込みラインおよび1つの読み取りラインを有する。第1のスイッチは、複数の入力と1つの出力を有し、前記メモリセルの1つの前記読み取りラインを前記読み取りポートに結合し、前記メモリセルの1つの前記書込みラインを前記書込みポートと結合させるための第2のスイッチが備えられる。

もう1つの本発明の目的は、速いコンテキスト切り替えオペレーションのための方法を提供することである。この目的は、少なくとも2つのメモリセルと1つのスイッチを含む少なくとも1つのレジスタを使うことによって、達成される。このスイッチは、メモリユニット内に蓄積された命令を実行するデータ処理ユニット内のメモリセルの1つのリード/ライトアクセスを行う。この方法は、命令あるいは例外処理のような、第1の前もって決定されたイベントの実行の際に、前記レジスタ内の第1のメモリセルを選択し、第2の前もって決定されたイベントの実行の際に前記レジスタ内の前記第1のメモリセルから前記レジスタ内のもう1つのメモリセルまで切り替えて、そして前記第1のメモリセルに切り替え戻しするステップを含んでいる。

コンテキストスイッチは、割り込み、またはトラップ、あるいは類似のイベント、の実行のような何らかのコールあるいはリターン命令いずれかによって起こる。コンテキストが1組のレジスタの内容から成ることがある。1つのコンテキストがいくつかのパーツ、例えば2つのパーツ、すなわち上位および下位コンテキスト、に分けられることができる。このような場合には、単に基本的なコンテキストを形成している1方の部分をセーブすることだけが求められるのがほとんどである。けれども、ある種の、いっそう複雑なルーチンは、基本的なコンテキストよりさらに多くをセーブする必要があるかもしれない。これらの実例として

は、特別な命令がコンテキストの他の部分をセーブ／復元させるための能力を提供する。本明細書においては、アプリケーションコンテキストは、コンテキストの基本パートあるいは完全なコンテキスト、の両方とも意味することができる。

基本的に、本発明はたいていの好ましい条件の下で、メモリから、またはメモリに、レジスタ内容の転送を必要としないような、コンテキストを自動的にセーブして、そして復元するメカニズムを提供する。コンテキストの一部を蓄積するそれぞれのレジスタは、2つの、あるいはもっと多くのメモリセルを含んでいる。それぞれのレジスタの内容をセーブして、そして復元する代わりに、本発明は適切な方法でそれぞれのレジスタ内の異なったメモリセル間を切り替えるスイッチングユニットを提供する。換言すれば、それぞれのコンテキストは、それぞれのレジスタ内の異なったメモリセルに割り当てられる。

それほどハードウェアに集約的でない実施例は、コンテキストに割り当てられているレジスタの一部分のみをセーブして、そして復元する。そのようにして、それぞれのコールあるいはリターン命令に伴った実行サイクルの数が節減される。

本発明による二重ビットのレジスタを有するレジスタファイルは、インテジャパイプラインオペレーションによってコールおよびリターンが並列に発せられることを可能にするという利点を有している。本発明はあらゆる種類のマイクロプロセッサあるいはマイクロコントローラに適用することができる。例えば、コール／リターンあるいはリンクされたリストのためのスタックを使用するマイクロプロセッサは、本発明から利益を得ることができる。統計的には、ほとんどのコールが、もう1つのサブルーチンへのさらなるコールなしで戻っている。これらの場合、従来の技術ではすべての必要なレジスタをセーブするために必要とされる多数のサイクルに比較して、1つのコールあるいはリターン命令が最低量のサイクルを必要とするだけである。

図面の簡単な説明

図1は、本発明によるマイクロプロセッサのレジスタファイルを表現しているブロック図を示している。

図2は、本発明によるレジスタファイルにおける、単独のレジスタセルを表現

しているブロック図を示している。

図3は、本発明による二重ビットレジスタセルのための2状態モデル図を示している。

図4は、本発明によるレジスタファイルにおける単独のレジスタセルの、もう1つの実施例を示している。

図5は、本発明による二重ビットレジスタセルのための4状態モデル図を示している。

図6は、本発明によるレジスタファイルにおける単独のレジスタセルのさらに別の実施例を示している。

図7は、本発明による二重ビットレジスタセルのための、もう1つの4状態モデル図を示している。

図8は、本発明による多数のセルを有する単独のレジスタの第3の実施例を示している。

発明の詳細な説明

図1は、例えば32ビットマイクロプロセッサの、あるいは32ビットマイクロコントローラの、ブロック図を示している。これらはコンテキスト切り替え命令を実行するためにリンクリストメカニズムを使用している。中央処理装置(CPU)100は、命令メモリ300と組み合わせられる命令制御ユニット101を含んでいる。この実施例では、命令制御ユニットは2つのパイプラインと、インテジャ実行ユニット102と、そしてロード/ストアユニット103とに供給する。インテジャ実行ユニット102は、全体的にデータレジスタファイル104と結びつけられ、そしてロード/ストアユニット103は全体的にアドレスレジスタファイル105と結合される。レジスタファイル104および105には、それぞれ8つの偶数の、および8つの奇数のレジスタが配置されていて、32の汎用レジスタを定義している。それぞれのレジスタはマルチポートアクセスを可能にしている。特に、偶数の、そして奇数のレジスタの配置は、それぞれのレジスタファイル内の2つのレジスタへの並列アクセスを可能にする。両方のパイプラインは共に、データレジスタおよびアドレスレジスタにアクセスすることができるけれども、このアクセスパスは、より良い理解を得るために図1には示さ

れていない。コンテキスト制御ユニット107が、両方のレジスタファイル104および105のそれぞれのレジスタ内の異なるメモリセルへのアクセスをコントロールする。多重システムレジスタ106は、CPU100内の種々の機能を制御する。この実施例におけるシステムレジスタ106の2つのレジスタPSWおよびPCXIは、リンクリストコンテキスト切り替えのために適切であって、そしてそれらは本発明による二重ビットレジスタであることが望ましく、それらの内容はこの実施例におけるコンテキストの一部であるようにされている。コンテキストスイッチ制御ユニット107は、例えば、状態マシンの手段によって、3つのレジスタファイル104、105、および106の異なるレジスタを制御する。それらのレジスタについては、以下にもっと詳細に説明される。例えばキャッシュメモリであるデータメモリ200は、データおよびアドレスレジスタファイル104および105と結合される。周辺バス500は、どのような周辺装置400でも、命令およびデータメモリ300および200およびCPU100と結びつけるために提供されている。

多くのマイクロプロセッサでは、1つからすべてのレジスタへの内容によって定義されるコンテキストは、個別の命令を通して蓄積されなくてはならない。本発明によれば、このプロセスは高度に自動化される。セーブ／蓄積して、そしてロード／復元することは、それぞれの第1そして第2のイベントにより自動的に実行される。このイベントは命令あるいは例外処理あるいはトラップあるいは類似のイベントであり得る。コンテキストは、例えば、レジスタD8…D15、およびA8…A15のようなレジスタファイルの多くのレジスタから成り立つことができる。この説明を通して、いわゆるリンクリストは複数の異なったコンテキストをセーブするために使われる。にもかかわらず、コンテキストの自動的な蓄積／復元することの基本原理は、他のいかなるメカニズム、例えばよく知られているスタックメカニズム、にでも容易に適合させることができる。説明された実施例では、コンテキストを蓄積／復元することは、データレジスタファイルのレジスタD8、D9…D15を、アドレスレジスタファイルのA10、A11…A15、そしてシステムレジスタのPCXIおよびPSWを蓄積／復元することを意味している。これらの16のレジスタは上位コンテキストを形成する。いずれ

かのルーチンが、付加的に他の16のデータおよびアドレスレジスタファイルのレジスタD0…D7、およびA0…A7を蓄積／復元する必要があるかもしれない。これらは下位コンテキストを構成する。特別な命令がこれを達成するために用意されている。にもかかわらず、同じ方法でこのオペレーションを速めるためには、これらのレジスタは本発明による同じ型式のものであり、そして同じ蓄積／復元メカニズムが適用される。これらについては以下にもっと詳細に説明されるであろう。

図2、図4、図6および図8によるすべてのレジスタは、より良い概観を得るために、ただ1つだけのビットを示している。もちろん、レジスタのサイズはどんな数のビットでもあり得る。次の事例では、32ビットのレジスタが説明される。

図2に示される1つだけのレジスタは、データレジスタファイル104における、そしてアドレスレジスタファイル105およびタ2つのシステムレジスタPSWおよびPCXIにおけるレジスタを表しており、それらはこの実施例では、タスクのコンテキストを蓄積するために使われている。例えば、データレジスタファイル104およびアドレスレジスタファイル105は、それぞれ16の32ビット幅のレジスタを含むことができる。データレジスタファイル104の上位の8つのレジスタおよびアドレスレジスタファイル105の上位の6つのレジスタもまた、2つのシステムレジスタであり、それらはタスクコンテキストの内容を蓄積するために使われる。もし必要であるなら、より低位の8つのレジスタが追加形のコンテキストレジスタとして用いられることもできる。残りのアドレスレジスタA8およびA9はグローバルポインタとして用いられる。そのために、この実施例においては、データレジスタファイルの少なくとも上位の8つのレジスタおよび、アドレスレジスタファイルの上位の6つのレジスタならびに、システムレジスタ106のPCXIそしてPSWレジスタは、図1によるレジスタであり、そしてそれぞれのレジスタファイルにおける他の10のレジスタは、従来技術に従ったレジスタであっても良い。もちろん、レジスタファイルの中のすべてのレジスタは本発明による二重ビットレジスタであることもできる。コンテキスト切り替えに責任を有しているシステムレジスタのいくらかもまた、本発明に

よる二重ビットのレジスタとして形成されることもできる。これは、それぞれのシステムがコンテキスト切り替えにどのように対処するかによっている。コンテキストをストアして、そしてロードする1つのメカニズム、いわゆるリンクリストメカニズム、は合衆国特許出願書第08/92852号「Data processing unit with hardware assisted context switching capability」で明らかにされている。その特許は本明細に参照されて含まれている。

図2は、例えば、図1で示されたようなマイクロプロセッサあるいはマイクロコントローラのレジスタファイルにおける1つのマルチポートレジスタセルを示している。この1つのレジスタは、第1のビットAおよび第2のビットBを蓄積するための2つのメモリセル1および2を含んでいる。両方のメモリセル1および2は、読み取りおよび書き込みポートを含んでいる。読み取りポートは、制御ライン30によってコントロールされるスイッチ3と結合される。スイッチ3は3つの制御可能なドライバ4、5および6のインプットに接続されている1つのアウトプットを含んでいる。ドライバ4、5および6のアウトプットは、それぞれ読み取りポートデータライン38、39および40に接続されている。ドライバ4、5および6の制御入力は、それぞれ読み取りポートワードライン33、34および35に接続されている。メモリセル1および2の書き込みポートは、書き込み制御論理29と結合されている。この書き込み制御論理29はまた、2つの書き込みポートデータライン36および37に結合される。書き込み制御論理29の制御入力、制御線30に、および書き込みポートワードライン31および32に結合される。書き込み制御論理29およびスイッチ3は、両方のメモリセル1あるいは2の1つを、それぞれの読み取りまたは書き込みポートデータラインに結びつけている。こうして、メモリセル1あるいは2の一方を現行のメモリセルとし、そして他を代替セルにする。

図2による設計は完全にスタティックであり得る。そのために、プレチャージされたビットラインあるいはセンスアンプは必要ではない。メモリセル1および2の読み取りポートの上には、マルチプレクサ3があり、2つの蓄積されたビット値の1つをポートライン上のトライステートドライバ4、5および6に渡す。

マルチプレクサ3は、ひとつのグローバルビット選択ライン30によってコントロールされている。同じラインはまた、書き込み制御論理29へのインプットであって、そしてメモリセル1あるいは2のいずれが書き込みポートによってアクセスされるかを選択する。この実施例では全体的に、1度にはただ1つだけのセルが「アクティブ」で、そしてポートのいずれを通してでもアクセス可能である。

図2による二重ビットのレジスタは、異なった方法でも使われることができる。これは以下にもっと詳細に説明されるであろう第1の2状態モデルが図3に示されている。文字Cが、コールあるいはコンテキストセーブを表し、文字Rはリターンまたはコンテキスト復元を表している。コンテキスト制御ユニット107は、この状態マシンを含んでいて、切り替えユニット29および3がそれぞれのレジスタ内の2つの異なったメモリセル1および2にアクセスする作用をコントロールする。

もし最後のオペレーションが文字Cによって示されるコールあるいはコンテキストセーブであったなら、代替のメモリセル2は、まだメモリにセーブされない以前のコンテキストを有している。もし次のオペレーションがリターンあるいはコンテキスト復元であるなら、この復元はただ、代替のセットのメモリセルにスイッチバックすることによって実行される。これは遷移 $R_1(0)$ によって示されている。レジスタデータ転送のためのメモリサイクルは必要とされない。これは R_1 遷移の後の括弧内の0によっても表現されている。他方しかしながら、もし次のオペレーションがコールであるなら、どれが現行の使用中のメモリセルであるかによって、正規の、あるいは代替のメモリセル1あるいは2内のセーブされていない以前のコンテキストは、このセットがコールされた機能のためのレジスタの新しいメモリセルになる前に、遷移 $C_2(4)$ を通してメモリに200に書かれなくてはならない。もしバス600が、例えば128ビット幅であるなら、4つのレジスタが並列にメモリ内にセーブされ得るであろう。この場合このように、8つの32ビット幅のデータレジスタおよび8つの32ビット幅のアドレスレジスタからなるコンテキストをセーブするためには4サイクルが必要とされる。このことは、 C_2 遷移の後の括弧内の4によって表現されている。

もし最後のオペレーションが図3における文字Rによって示されるリターンあ

るいはコンテキスト復元であったなら、代替のメモリセルは終了したコンテキストからのデッドな値を有している。この場合には、もし次のオペレーションがコールであるなら、コンテキストセーブはただ制御線30を通して、代替のメモリセル内に「セーブされた」コンテキスト内容を残して、ビット選択をトグルすることによって実施される。これは遷移 $C_1(0)$ によって示されている。メモリデータ転送のためには、レジスタのサイクルが必要とされない。他方、もし次のオペレーションがもう1つのリターンであるなら、復元されたコンテキストは遷移 $R_2(4)$ によってメモリ200からもって来られなくてはならない。再びコンテキストを読み取ることは、128ビット幅のバスで4サイクルをとる。

コールが高度の確率でリターンに変わるときには、この2ステートの実施例はよく作動する。しかしながら、異なったアプリケーションおよびそれぞれのプログラムは、より深いスタックされたルーチンを構築する一連のより多くのコールを有することもある。

そのために、図5では、コンテキスト切り替えをコントロールするための4状態マシンが描写されている。この4状態マシンは、よりいっそう進歩していて、そしてそれで、4サイクルコールとリターンのパーセンテージを実質的に引き下げる。それは例えば、2状態モデルがセーブまたは復元サイクルを必要としない状況の下でさえ、代替のメモリセル内にコンテキストの2分の1を送り込むような、きわどいセーブまたは、復元をすることによって実行される。それで例えば、ただ制御線30をトグルさせることによって、現行のコンテキストが代替のメモリセル内に「セーブされる」コールにおいて、トグルを行う前に、メモリ内にコンテキストの2分の1をきわどく書き込むのである。それから、もし次のオペレーションがもう1つのコールであることが分かったときには、代替のメモリセル内にコンテキストの2分の1をすでにセーブしてある。

類似の戦略がリターンのためにも使われる。フルのリターンコンテキストが代替のメモリセル内に維持されているリターン、すなわちコールの後に続くリターン、においては、代替のメモリセルへの切り替えは直ちには実行されない。その代わりに、代替のセットにトグルする前に、次のリターンコンテキストの2分の1は、現行のビットセット内にきわどく読み込まれる。それから、もし次のオペ

レーションがもう1つのリターンであったなら、新しいリターンコンテキストの2分の1は、すでにメモリ200から読み取られているので、ただ2つだけのサイクルがフルのコンテキストを復元することを完了するために必要とされる。

この4状態モデルは、コンテキストセーブ/復元アクティビティを「平準化」して、コールおよびリターンブランチの間に利用可能であるサイクルのより効果的な使用を推進する。最も良い結果を達成するために、コンテキスト制御装置107のための状態モデルは、コール/リターン命令の必要なサイクルに適合させられることができる。例えば、もしコール/リターン命令を実行するのに2つのサイクルを必要とするなら、4状態のモデルは適当な解法を提供する。本発明による、さらにいっそう複雑なマルチ状態モデルでさえ、速いコンテキスト切り替えを提供するために使われることができる。

この2状態モデルは、すべてのコールおよびリターンを2サイクルオペレーションで実行することができない。それは、一続きの2つのコールの後には、上位のレジスタセットの代替のメモリセルはフルのコンテキストを保持していて、それらはまだメモリ内に書き込まれてはいないからである。もし次のオペレーションがさらにもう1つのコールであるなら、それを保持しているメモリセルが次のコンテキストのために利用されるようになる前に、古いコンテキストを書き上げるために4サイクルが要求される。同じく、一続きの2つのリターンの後には、次のリターンコンテキストは代替のメモリセル内にまったくロードされない。3番目の連続したリターンは、128ビットのバスの上で、リターンコンテキストをロードするために、4サイクルを必要とするであろう。しかしながら、一続きの3つのコールあるいは3つのリターンのための統計値としては、それらは2つのコールあるいは2つのリターンよりもずっと多い。それらは普通、ゼロから最高でも6パーセントの間の範囲内にある。

そのために、この実施例では、4状態モデルは非常に適切である。低いひん度の理由は、かなり容易に理解することができる。たいいていのアプリケーションでは、一般にリーフ関数でない関数が多数のコールを実行する。まさしく1つのコールだけを実行することは、1つの関数にとっては普通ではない。多数のコールを実行する関数では、ただ最初のものだけが直接的にもう1つのコールの後に続

くであろう。残りはすべてリターンの後に続くであろう。それで、実質的なコール深さに達するアプリケーションにおいてさえ、一続きの3つのコールあるいは3つのリターンの統計上の頻度は低いという傾向がある。

この4状態モデルでは、内部状態の2ビットが拡張されたOPコードビットとして機能して、そしてコールおよびリターン命令それぞれを4状態依存の変形に分ける。しかしながら、コール変形の2つ、およびリターン変形の2つは、縮重されるものであり、他の言葉で言えば、それらは同じものにまで減じられる。それで実際には図4で示されている状態遷移図中において C_1 、 C_2 および C_3 とラベル付けされた3つのコール変形があり、そして R_1 、 R_2 および R_3 とラベル付けされた3つのリターン変形がある。これらの6つの命令変形のサイクルからサイクルへのオペレーションは、二重ビットレジスタのため異なる実施例に関連して以下に説明される。

図4が両方のモデルと一緒に使われることができる二重ビットレジスタの第2の実施例を示している。図1と同じ素子には同じ数字が伴っている。この実施例では、図1のマルチプレクサ3は2つのマルチプレクサ4 1および4 2で置き換えられている。両方のマルチプレクサ4 1および4 2のインプットは、メモリセル1および2の両方の読み取りポートに結合される。第1のマルチプレクサ4 1のアウトプットは、ドライバ4のインプットに接続されるのに対して、第2のマルチプレクサのアウトプットは、ドライバ5および6のインプットに接続されている。マルチプレクサ4 1は、第2の制御線4 3によってコントロールされ、そしてマルチプレクサ4 2は制御線3 0によってコントロールされる。

この配置は、読み取りポートが、グローバルな制御線3 0から独立して、メモリセル1あるいは2のいずれかのビットにランダムにアクセスすることを可能にする。それで、そのことはインテジャパイプ内のオペレーションと並列にコールが発せられることを可能にして、そしてコール当りの平均のサイクル時間をおよそ半分に下げるであろう。

図5は、例えば図4で示された二重ビットレジスタで使われることができる4状態モデルを示している。この円はコンテキストスイッチユニットが稼働する4つの異なった段階を示している。それぞれの文字の対が最後の2つの命令を示し

ている。換言すれば、CCは続いた2つのコールを表す。「一続きのコールあるいはリターン」または「コール／リターンシーケンス」は、この説明を通じて、どんなリターン／コールもない2つの連続したコール／リターンを意味している、しかしそれらの間に多くの他の非コンテキスト切り替え命令を有することは、理解されるはずである、CRはコール、リターンシーケンスであり、RCはリターン、コールシーケンスおよび、RRは2つの連続したリターンを表している。1つの状態から他への異なる遷移は以下に説明される。上位のコンテキストは、レジスタA10-A15、D8-D15と、そして2つのシステムレジスタ、すなわちプログラムステータスワードPSWおよびリンクレジスタPCXI、から成り立つと想定する。

C₁は、RRあるいはCR状態からRC状態へのコールによって起こされた遷移を示している。以下においては、メモリセル1が現行のメモリセルであり、そしてメモリセル2は代替のメモリセルであると仮定する。プログラムカウンタと、メモリロケーションおよび他のシステムレジスタを示すコンテキストスイッチレジスタとのセッティングのほかに、それぞれのマイクロ命令は基本的に、スイッチ41あるいは42を通して、レジスタD8、D9、PCXIおよびPSWのメモリセル1の内容をメモリ内にセーブする。続いてライン30がトグルされ、それによってドライバ5および6を他のメモリセル2と結び付ける。同時に、あるいは引き続いて、レジスタD10、D11、A10およびA11の現行のメモリセル1の内容がメモリ41を通してメモリ内にセーブされる。最終的に、ライン43がトグルされ、それによってドライバ4を他のメモリセル2と結び付ける。今や、メモリセル2は現行のメモリセルであり、そしてメモリセル1は代替のメモリセルである。

C₂は、RC状態からCC状態へのコールによって起こされる遷移を示している。以下ではメモリセル2が現行のメモリセルであり、そしてメモリセル1は代替のメモリセルであると仮定する。最初に、ライン43がトグルされ、メモリセル1をドライバ4と接続する。レジスタA12、A13、D12およびD13のメモリセル1の内容は、スイッチ41およびドライバ4を通してメモリにセーブされる。その後、あるいは並列に、ラインで30がトグルされ、そしてレジスタ

A14、A15、D14およびD15のメモリセル1の内容がスイッチ42を通してセーブされる。今や、メモリセル1はすべてのレジスタの現行のメモリセルであり、そして現在蓄積されているコンテキストは今、完全にセーブされている。

C₃はCC状態の中のコールオペレーションを示している。ここでは遷移は起きない。以下ではメモリセル1が現行のメモリセルであり、そしてメモリセル2が代替のメモリセルであると仮定する。今やメモリセル1および2の両方には、有効な、セーブされていないデータが存在する。従って、それ以上のコールを可能にするためには、完全なコンテキストがセーブされなくてはならない。最初に、ライン43がトグルされ、メモリセル2をドライバ4に切り替える。レジスタD8、D9、PCXIおよびPSWのメモリセル2の内容がスイッチ41およびドライバ4を通してメモリにセーブされる。ライン30がトグルされ、そしてレジスタA10、A11、A12、A13、A14、A15、D10、D11、D12、D13、D14、D15のメモリセル2の内容がメモリにセーブされる。

R₁は、RCまたはCC状態からCR状態へのリターンによって生じた遷移を示している。以下ではメモリセル1が現行のメモリセルであり、およびメモリセル2が代替のメモリセルであると仮定する。以前にセーブされたレジスタPSW、PCXI、D8およびD9の内容は、メモリ200から書込み制御論理29を通してそれぞれのレジスタの現行のメモリセル1にロードされる。それから、ライン30および43がトグルされ、それぞれのレジスタのメモリセル2を、書込み制御論理に、そしてドライバ4、5および6に切り替える。

R₂はCR状態からRR状態へのリターンによって生じる遷移を表している。以下ではメモリセル2が現行のメモリセルであり、そしてメモリセル1が代替のメモリセルであると想定する。最初に、ライン30および43がトグルされ、メモリセル1をドライバ4、5および6に接続する。レジスタA12、A13、D12およびD13のメモリセル1の内容が、メモリから書込み制御論理29を通して復元される。今やメモリセル1はすべてのレジスタの現行のメモリセルであり、そして前にセーブされたコンテキストは今、完全に復元される。

R₃は、RR状態の中のリターンオペレーションを示している。ここでは遷移

が起きない。以下ではメモリセル1が現行のメモリセルであり、そしてメモリセル2が代替のメモリセルであると想定する。1および2の両方のメモリセルが今「無効」データを有している。従って、それ以上のリターンを可能にするためには、完全なコンテキストが復元されなくてはならない。最初に、ライン30および43がトグルされ、メモリセル2を書込み制御論理29に切り替える。レジスタA10、A11、A12、A13、A14、A15、D8、D9、D10、D11、D12、D13、D14、D15、PCXIおよびPSWのメモリセル2の内容が、書込み制御論理29を通して復元される。

図6が本発明による二重ビットのレジスタの、さらに別の実施例を示している。同じ数字が同じ素子を示している。それぞれのメモリセル1および2が、2つのインバータによって構成されている。第1のインバータのアウトプットは、読み取りポートを形成しており、そして第2のインバータのインプットと結合されている。そして第2のインバータのアウトプットは書込みポートを形成しており、そして第1のインバータのインプットと結合されている。この実施例は、マルチプレクサスイッチ3と結合される読み取りポートサイドの上に、さらに別のドライバ7を提供する。ドライバ4、5、6および7のアウトプットは、リードポートデータライン21、22、23および24と結合される。ドライバ4、5、6および7の制御入力は、リードポートワードライン25、26、27および28に接続される。書込み制御論理は、3つの転送トランジスタカップルを含んでいる。トランジスタ対8、9；10、11；12、13は、2つのメモリセル1および2へのランダムアクセスを可能にする装置を構成する。3つの書込みポートワードラインカップルは、6つのゲートに接続されている。それらのゲートはトランジスタ8、…、13を個々にコントロールする。トランジスタ8および9は、書込みポートライン14をメモリセル1およびメモリセル2の書込みポートにつなぐことができ、トランジスタ10および11は、書込みポートライン15をメモリセル1およびメモリセル2の書込みポートにつなぐことができ、トランジスタ12および13は、ライトポートライン16をメモリセル1およびメモリセル2の書込みポートにつなぐことができる。選択ライン20は、マルチプレクサ3を制御する。このマルチプレクサは、ドライバ4、5、6、および7のイ

ンプットをメモリセル1あるいは2の読み取りポートのいずれかに接続する。

図7が、関連する4状態モデルを示している。この4状態モデルは、遷移に関しては本質的に図5に示された4状態モデルとまったく同じである。付加的に、遷移 $R_4(1)$ が実行される。図6による実施例に関するそれぞれの遷移によって行われるステップは、図4によるものとは少し異なっている。

ここでもまた、 C_1 は、RRあるいはCR状態からRC状態へのコールによって起こされた遷移を示している。以下ではメモリセル1が現行のメモリセルであり、そしてメモリセル2が代替のメモリセルであると想定する。プログラムカウンタと、メモリロケーションを示すコンテキストスイッチレジスタと、そして他のシステムレジスタをセットすることのほかに、それぞれのマイクロ命令は基本的に、レジスタD8、D9、PCXI、PSW、D10、D11、A10 およびA11のメモリセル1の内容を、マルチプレクサスイッチ3を通して、メモリにセーブする。それからライン20がトグルされて、ドライバ4、5、6および7を他のメモリセル2と接続する。今や、メモリセル2は現行のメモリセルであり、そしてメモリセル1が代替のメモリセルである。

C_2 はまた、RC状態からCC状態までのコールによって起こされた遷移を示している。以下ではメモリセル2が現行のメモリセルであり、そしてメモリセル1が代替のメモリセルであると想定する。最初に、ライン20がトグルされ、メモリセル1をドライバ4、5、6および7と接続する。レジスタA12、A13、A14、A15、D12、D13、D14およびD5のメモリセル1の内容がマルチプレクサスイッチ3を通してメモリにセーブされる。メモリセル1は今や、すべてのレジスタの現行のメモリセルであり、そして蓄積されたコンテキストが今、完全にセーブされている。

C_3 は再び、CC状態の中におけるコールオペレーションを示している。ここでは遷移が起きない。以下ではメモリセル1が現行のメモリセルであり、そしてメモリセル2が代替のメモリセルであると想定する。両方のメモリセル1および2は今や、有効な、しかもセーブされていないデータを有している。従って、それ以上のコールを可能にするためには、完全なコンテキストがセーブされなくてはならない。最初に、ライン20がトグルされ、メモリセル2がドライバ4、

5、6および7に切り替えられる。レジスタD8、D9、PCXI、PSW、A10、A11、A12、A13、A14、A15、D10、D11、D12、D13、D14およびD15のメモリセル2の内容は、マルチプレクサスイッチ3を通してメモリにセーブされる。

R_1 は、RCあるいはCC状態からCR状態へのリターンによって引き起こされた遷移を示している。以下ではメモリセル1が現行のメモリセルであり、そしてメモリセル2が代替のメモリセルであると想定する。以前にセーブされたレジスタPSW、PCXI、D8およびD9の内容は、メモリ200から、転送トランジスタ8、10、あるいは12の1つを通して、それぞれのレジスタの現行のメモリセル1までロードされる。続いて、ライン20がトグルされ、アウトプットを、それぞれのレジスタのメモリセル2に切り替える。

同じく、 R_2 はCR状態からRR状態までのリターンによって起こされた遷移を示している。以下ではメモリセル2が現行のメモリセルであり、そしてメモリセル1が代替のメモリセルであると想定する。最初に、ライン20がトグルされ、メモリセル1をアウトプットドライバ4、5、6および7に接続する。レジスタA12、A13、D12、D13、A14、A15、D14およびD15のメモリセル1の内容が、メモリから、転送トランジスタ8、10あるいは12の1つを通して復元される。メモリセル1は今や、すべてのレジスタの現行のメモリセルであり、そして前にセーブされたコンテキストは今、完全に復元されている。

R_3 は、RR状態中でのリターンオペレーションを示している。再びここでは、遷移が起きない。以下ではメモリセル1が現行のメモリセルであり、そしてメモリセル2は代替のメモリセルであると想定する。両方のメモリセル1および2は、今や「無効な」データを有している。従って、それ以上のリターンを可能にするためには、完全なコンテキストが復元されなくてはならない。最初に、ライン20がトグルされ、それによってメモリセル2をアウトプットドライバ4、5、6および7に切り替える。レジスタA10、A11、A12、A13、A14、A15、D8、D9、D10、D11、D12、D13、D14、D15、PCXIおよびPSWのメモリセル2の内容は、メモリから、転送トランジスタ

9、11、あるいは13を通して復元される。

R₄は、RC状態からのリターンによって起こされた遷移を示している。そこでは、リターンコンテキストがコールチェーンのベースである。この遷移に関しては、メモリからメモリセルへの転送が起きないため、いずれのメモリセルが現行のものであるかは重要ではない。単に、ライン20がトグルされ、他のメモリセルへ切り替えられるだけである。

図6による実施例は、いずれのビットが書き込みポートから書きこまれるかを選択するために、個々のワードライン17、18および19を使う。さらに2本の制御線がセルを横切って走っているのではあるが、この設計は、いずれのメモリセルが書かれるかをコントロールするために、ただ3つだけの書き込みポートワードラインおよびステアリングロジックを有する場合に比して、より小さくて、より速くて、そしていっそう信頼性が高い。個々のワードライン17、18および19を用いると、ひとつのn-トランジスタが、書き込みデータラインと、選択されたメモリセルとの間のゲーティングのために使われることができる。もしストレージメモリセルが書き込みポートワード選択の下流に選択されたなら、それぞれのメモリセルがトランスミッションゲートによって前渡しされる必要があるであろう。それは電荷シェアリング問題を起こし、そしてクロック遷移タイミング感度問題を提出するであろう。1つのローカルなインバータが、トランスミッションゲートへのライン20上に信号を提供するために必要とされるであろう。

この設計におけるビットセルのランダムな書き込みアドレス指定能力は、都合がよい解法を提供する。なぜなら次の命令の前のR1シーケンスにおけるサイクルの数はディスパッチされることができるからである。R1シーケンスでは、ライン20がトグルされて、リターンコンテキストにおいてレジスタのデコードサイクル読み取りを可能とするまでは、新しい命令を出すことができない。この実施例では、読み取り、そして書き込みすることは独立している。そのために書き込みシーケンスの間にライン20をトグルするためのタイミングは重要ではない。この実施例では、制御線20は、次の命令を出すことを可能にするために、早目にトグルされることができる。他方、次のコンテキストのプレロードは、書き込みポート上のランダムアドレス能力を使って継続される。

図8は複数のメモリセル1 a、1 b、…、1 nを有する単独のレジスタを示している。この実施例は、書き込み制御論理2 9およびマルチプレクサスイッチ3を提供する。このスイッチは複数の制御線3 0 a、…、3 0 mによってコントロールされる。ここにおいて、制御線のナンバー>= 1 d (メモリセルの数)である。それぞれのコールを用いて、コンテキストスイッチ制御ユニットが次の利用可能なメモリセルに切り替える。例えばメモリセル1 nからメモリセル1 aまでのスイッチの中で、さらなるメモリセルが利用不可能な場合にだけは、新しいメモリセルが使われることができる前に、次のメモリセルの内容がメモリ内にセーブされなくてはならない。これは例えば、トラップの手段によって、あるいは上記で説明された方法の1つの手段によって、容易に実行することができる。このような装置は、メモリセルの数に依存して、レジスタファイルとメモリとの間の転送を伴わないで、あるいは最小限だけを用いて、異なるコンテキストを切り替える最も速い方法を提供する。

もし異なるタスクのそれぞれが、コンテキストを有しており、そのコンテキストが異なったレジスタを含んでいて、そして従って異なったサイズを有しているなら、特別な関数レジスタが提供されるかもしれない。このレジスタはレジスタファイル1 0 4および1 0 5のいずれのレジスタがセーブされるかを示すものである。レジスタファイル1 0 4および1 0 5の両方が共に、3 2のレジスタを含むなら、システムレジスタファイル1 0 6における特別な3 2ビットのレジスタが、それぞれのビットによって、いずれのレジスタがそれぞれのコンテキストに割り当てられているかを示すことができる。下位ビット0から1 5は、データレジスタを、および上位ビット1 6から3 1までは、アドレスレジスタを示すことができる。システムレジスタファイルにおけるこのようなレジスタもまた、本発明によるマルチビットのレジスタであることが望ましい。それらのレジスタもまた、それぞれのコンテキストの一部分である。そのようなコンセプトは非常にフレキシブルなメカニズムを提供するとは言っても、異なるレジスタを取り扱うためには、またさらに複雑な制御メカニズムを必要とする。例えばこのコンセプトは、すべての状況の下でそれぞれのコンテキスト切り替えルーチンを速めることができる汎用レジスタを必要としない特定のコンテキストには、何のレジスタも

割り当てないという能力を提供する。

システムレジスタファイル内でこのようなレジスタを使っているもう1つの実施例は、上に説明されたコンセプトに類似しているそれぞれのタスクに割り当てられている固定された数のレジスタを使うことができる。例えば、特殊レジスタの手段によってコンテキストを定義するために16のレジスタが使われることができる。それぞれのコンテキストのために同じ16のレジスタを使う必要はない。ユーザは32のレジスタから16を選択して、そしてシステムレジスタファイル内の特殊レジスタの手段によってそれらを割り当てることができる。

【図面の簡単な説明】

【図1】

図1は、本発明によるマイクロプロセッサのレジスタファイルを表現しているブロック図を示している。

【図2】

図2は、本発明によるレジスタファイルにおける、単独のレジスタセルを表現しているブロック図を示している。

【図3】

図3は、本発明による二重ビットレジスタセルのための2状態モデル図を示している。

【図4】

図4は、本発明によるレジスタファイルにおける単独のレジスタセルの、もう1つの実施例を示している。

【図5】

図5は、本発明による二重ビットレジスタセルのための4状態モデル図を示している。

【図6】

図6は、本発明によるレジスタファイルにおける単独のレジスタセルのさらに別の実施例を示している。

【図7】

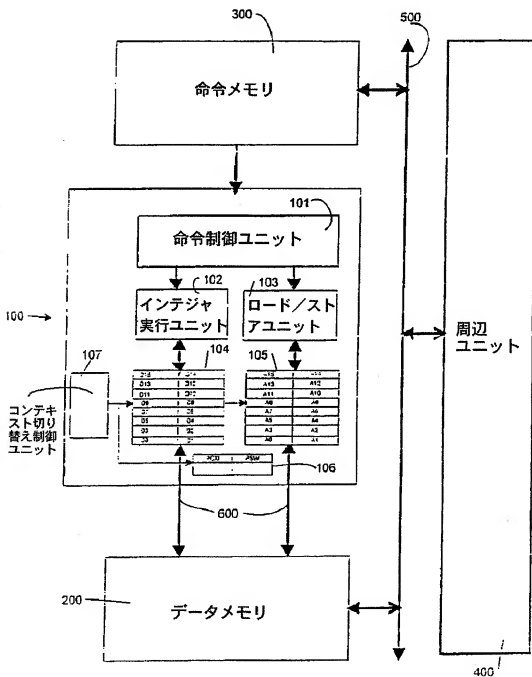
図7は、本発明による二重ビットレジスタセルのための、もう1つの4状態モ

デル図を示している。

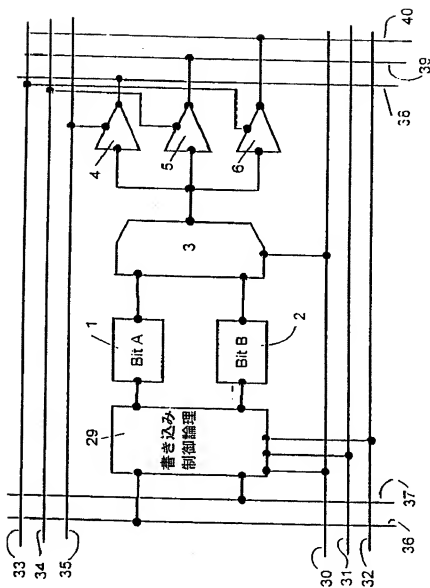
【図 8】

図 8 は、本発明による多数のセルを有する単独のレジスタの第 3 の実施例を示している。

【図 1】



【図2】



【図3】

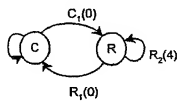
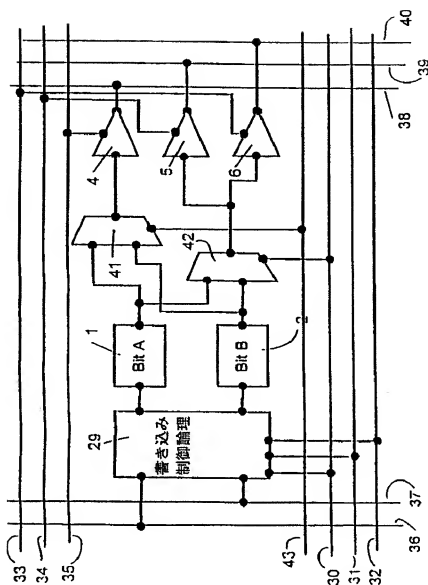


Fig. 3

【図4】



【図5】

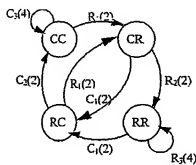


Fig. 5

【図 6】

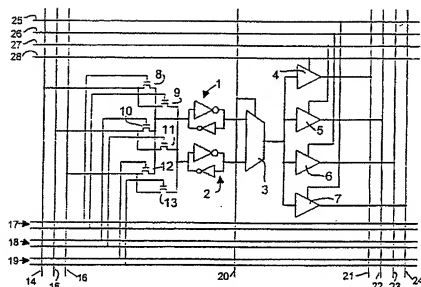


Fig. 6

【図 7】

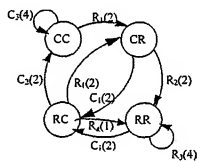
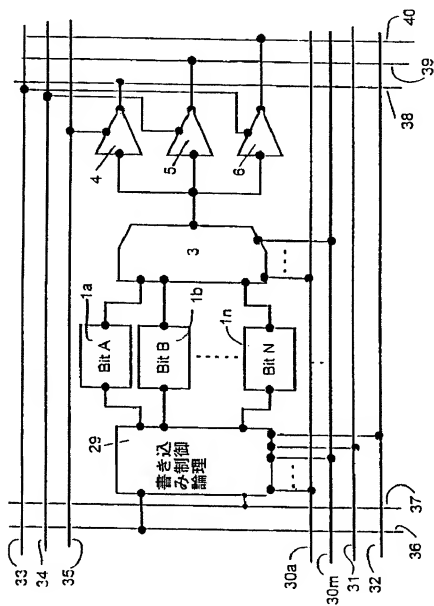


Fig. 7

【図8】



【手続補正書】特許協力条約第34条補正の翻訳文提出書

【提出日】平成12年5月24日(2000. 5. 24)

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】特許請求の範囲

【補正方法】変更

【補正内容】

【特許請求の範囲】

【請求項1】 データ処理ユニットにおいて、

少なくとも1つの読み取りポートと1つの書き込みポートを有する、少なくとも1つのレジスタを含み、

前記レジスタは少なくとも2つのメモリセル(1、2; 1a、1b、...、1n)を有し、

メモリセルのそれぞれは1つの書き込みラインと、1つの読み取りラインを有し、

前記メモリセル(1、2; 1a、1b、...、1n)の1つの前記読み取りラインを、前記読み取りポートに結合させるための、複数のインプットおよび1つのアウトプットを含む第1のスイッチ(3; 41、42)を含み、

前記メモリセル(1、2; 1a、1b、...、1n)の1つの前記書き込みラインを、前記書き込みポートに結合させるための、少なくとも1つのインプットおよび複数のアウトプットを有する第2のスイッチ(29; 8、9、10、11、12、13)を含む、ことを特徴とするデータ処理ユニット。

【請求項2】 多数の読み取りポートドライバ(4、5、6; 7)および複数の読み取りポートデータライン(38、39、40; 21、22、23、24)をさらに含み、

それぞれの読み取りポートドライバ(4、5、6; 7)が、1つのインプットと1つのアウトプットを有し、

それによって、前記読み取りポートドライバ(4、5、6; 7)の前記インプットが、前記第1のスイッチ(3; 41、42)の前記アウトプットに接続され

前記読み取りポートドライバ(4、5、6、7)の前記アウトプットが、前記読み取りポートデータライン(38、39、40:21、22、23、24)に接続されている、請求項1に記載のデータ処理ユニット。

【請求項3】 複数の書き込みポートデータライン(31、32;17、18、19)をさらに含み、

前記第2のスイッチ(29;8、9、10、11、12、13)が、それぞれの書き込みポートデータライン(31、32;17、18、19)を、それぞれのメモリセル(1、2;1a、1b、...、1n)の前記書き込みラインに結合させることができる、請求項1または2に記載のデータ処理ユニット。

【請求項4】 前記第2のスイッチ(29;8、9、10、11、12、13)が、それぞれのメモリセル(1、2;1a、1b、...、1n)へのランダムアクセスを提供する、請求項1から3のいずれか1つに記載のデータ処理ユニット。

【請求項5】 前記第2のスイッチが、複数の転送トランジスタ(8、9、10、11、12、13)を含み、

それらのトランジスタは、それぞれのメモリセル(1、2)のそれぞれの書き込みラインを、前記書き込みポートの1つに結び付ける、請求項4に記載のデータ処理ユニット。

【請求項6】 前記レジスタの内容を蓄積し、そしてロードするための制御ユニット(107)をさらに含む、前出請求項のいずれか1つに記載のデータ処理ユニット。

【請求項7】 前記制御ユニット(107)が状態マシンを含む、請求項7に記載のデータ処理ユニット。

【請求項8】 データ処理ユニットにおいて、

第1のレジスタ(104)のセットと、そして第2のレジスタ(105)のセットを有する、レジスタファイル(104、105)を含み、

前記第1のレジスタ(104)のセット内の前記レジスタは、少なくとも1つの読み取りポートと、1つの書き込みポートを有し、それぞれが2つのメモリセル

(1, 2; 1a, 1b, . . . , 1n) を有し、それぞれが書込みラインおよび読み取りラインを有し、

前記第1のレジスタ(104)のセットのそれぞれのレジスタの前記メモリセル(1, 2; 1a, 1b, . . . , 1n)の1つの前記読み取りラインを、それぞれの読み取りポートに結合させる、第1のスイッチ(3; 41, 42)を含み、

前記第1のレジスタ(104)のセットのそれぞれのレジスタの前記メモリセル(1, 2; 1a, 1b, . . . , 1n)の1つのそれぞれの書込みラインを、前記書込みポートに結合させるための、第2のスイッチ(29; 8, 9, 10, 11, 12, 13)を含む、

ことを特徴とするデータ処理ユニット。

【請求項9】 前記レジスタ(104, 105, 106)がそれぞれ多数の読み取りおよび多数の書込みポートを含み、

前記第1そして第2のスイッチ(3, 29; 41, 42; 8, 9, 10, 11, 12, 13)が、前記多数の読み取りおよび書込みポートを、前記第1のレジスタ(104)のセットのそれぞれのレジスタのそれぞれのメモリセル(1, 2; 1a, 1b, . . . , 1n)に結合させる、請求項8に記載のデータ処理ユニット。

【請求項10】 第1のレジスタ(104)のセットの前記レジスタの内容を蓄積し、そしてロードするための制御ユニット(107)をさらに含む、請求項8または9に記載のデータ処理ユニット。

【請求項11】 前記制御ユニット(107)が状態マシンを含む、請求項10に記載のデータ処理ユニット。

【請求項12】 前記第1のレジスタ(104)のセットが、データおよびアドレスレジスタを含む、請求項8から11のいずれか1つに記載のデータ処理ユニット。

【請求項13】 請求項1から12のいずれかに記載のデータ処理ユニット内で第1のタスクから第2のタスクへのコンテキスト切り替えを行う方法において、

a) 第1の前もって決定されたイベントの実行によって、前記レジスタ(104, 105)のそれぞれにおいて次の利用可能なメモリセル(1, 2; 1a, 1b, . . . , 1n)の読み取りポートに切り替えるステップと、

b) 第1の前もって決定されたイベントの実行によって、そしてもう1つのメモリセル(1, 2; 1a, 1b, . . . , 1n)に切り替える前に、前記メモリセル(1, 2; 1a, 1b, . . . , 1n)の中の前記レジスタのセットのレジスタの前もって定められた数の現行のメモリセル(1, 2; 1a, 1b, . . . , 1n)の内容を蓄積するステップと、

c) 第2の前もって決定されたイベントの実行によって、前記レジスタ(104, 105)のそれぞれにおける以前のメモリセル(1, 2; 1a, 1b, . . . , 1n)に切り替え戻しするステップを含む、ことを特徴とするコンテキスト切り替えを行う方法。

【請求項14】 第1の前もって決定されたイベントの実行によって、もう1つのメモリセル(1, 2; 1a, 1b, . . . , 1n)に変わる前に、前記レジスタのセットの残っているレジスタの数の現行のメモリセル(1, 2; 1a, 1b, . . . , 1n)の内容を、前記メモリセル(1, 2; 1a, 1b, . . . , 1n)の中に蓄積するステップを含む、請求項13に記載の方法。

【請求項15】 前記第2の前もって決定されたイベントの実行によって、もう1つのメモリセル(1, 2; 1a, 1b, . . . , 1n)に切り替わった後で、

前記レジスタのセットの残っているレジスタの数の蓄積されているメモリセル(1, 2; 1a, 1b, . . . , 1n)の内容を、前記メモリセル(1, 2; 1a, 1b, . . . , 1n)から前記レジスタ(104, 105)の中にロードするステップを含む、請求項14に記載の方法。

【請求項16】 前記第2の前もって決定されたイベントの実行によって、もう1つのメモリセル(1, 2; 1a, 1b, . . . , 1n)に切り替わる前に、

前記レジスタのセットの前もって定められた数のレジスタの蓄積されているメモリセル(1, 2; 1a, 1b, . . . , 1n)の内容を前記メモリセル(1,

2 ; 1 a、1 b、...、1 n) から前記レジスタ (1 0 4、1 0 5) の中にロードするステップを含む、請求項 1 3 に記載の方法。

【手続補正 2】

【補正対象書類名】明細書

【補正対象項目名】0 0 0 1

【補正方法】変更

【補正内容】

【0 0 0 1】

発明の背景

本発明はコンテキスト切り替え能力を有する装置に関する。ほとんどの埋め込み形の、そして実時間のコントロールシステムは、割込みハンドラおよびソフトウェアによって管理されたタスクがそれぞれ、それら自身の仮想のマイクロコントローラ上で実行されるよう、考慮されているモデルに従って設計されている。そのモデルは一般に実時間実行あるいはオペレーティング・システムのサービスによってサポートされている。それらは基礎をなしているマシンアーキテクチャの特徴および能力のトップ上に層をなしている。仮想のマイクロコントローラは、それ自身の汎用レジスタと、そしてプログラムカウンタのような関連する特殊機能レジスタと、プログラムステータスワードなどを有する 1 つのタスクとして見ることができる。そしてそれらはこのタスクのコンテキストを表している。公知のシステムの大部分における、これらの仮想のマイクロコントローラの取り扱いは、それぞれのコンテキストをセーブして、そして復元するソフトウェアの手段によって実行される。そのために、このようなデータ処理ユニットのためのソフトウェアは、増加する一方の量のメモリを必要とする。そしてコンテキスト切り替えオペレーションのための実行オーバーヘッドはアプリケーションタスクに利用可能な処理帯域幅を減らす。

一般に、コンテキストスイッチはレジスタファイルの少なくともある特定のレジスタの内容が、前もって定められたメモリエリア内に蓄積されて、そしてもう 1 つのメモリエリアの内容で置き換えられることを必要とする。それによって、それぞれのメモリエリアが、特定のタスクあるいは割り込みサービスルーチンと

関係するすべての必要なデータを含んでいる特定のコンテキストを表す。コンテキスト切り替えと呼ばれるこのオペレーションは、いわゆる実時間オペレーティング・システムソフトウェアの中で決定的に時間を消費する部分である。このようなオペレーティング・システムはしばしば、外部か、あるいは内部のイベント上で可能な限り速く反応しなくてはならない、という目的を有する。

第US-A-4410939号は、中央処理装置に含められる複数のレジスタを開示している。それらのレジスタは、いくつかのグループまたはセットに分けられて、割り込みサービスのための応答時間を改善する。レジスタセットのすべては、同じ構造を有する。割り込み要求に応答するために、ただ1つだけのレジスタセットが、前もって決定されたシーケンスにおいて次々にイネーブルとされる。より高い優先順位レベルの割り込みサービスルーチンが完了されると、レジスタセットの1つが逆順で次々にイネーブルとされる。すべてのレジスタセットがロードされたとき、最も古いデータを蓄積している1つのレジスタセットの内容が、メインメモリのセーブ用領域内にセーブされることができる。セーブされたデータは逆方向で検索することもできる。

第GBA-2 216 307号は、コンピュータシステムが比較的大量のベクトルレジスタファイルスペースを有するように構成することを可能とするために、SRAMの使用を開示している。このスペースは複数のコンテキストエリアに分けられることもでき、それぞれのコンテキストエリアは個々のプロセスをサポートしている。このようなコンテキストエリアの使用は、プロセス間を切り替えるときに、ベクトルレジスタファイルの中のコンテキスト情報をディスクにスワップする必要を退ける。

【国際調査報告】

INTERNATIONAL SEARCH REPORT		Inventor's Application No. PCT/US 99/04134
A. CLASSIFICATION OF SUBJECT MATTER IPC 6 G06F9/46 G11C7/00		
According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) IPC 6 G06F G11C		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
Electronic data base consulted during the international search (name of data base and, where practical, search terms used)		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y A	US 4 410 939 A (KAWAKAMI KATSURA) 18 October 1983 (1983-10-18) abstract column 3, line 18 - last last column 10, line 44 - column 11, line 9; claims 1,2 — — — — — -/-	13, 14, 17, 18, 21-23, 26, 27, 30 1-12, 15, 16, 19, 20, 24, 25, 28, 29
<input checked="" type="checkbox"/> Further documents are listed in the continuation of box C. <input checked="" type="checkbox"/> Patent family members are listed in annex.		
* Special categories of cited documents: "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier document but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as appearing) "O" document relating to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "Z" document member of the same patent family		
Date of the actual completion of the international search 30 July 1999		Date of mailing of the international search report 13/08/1999
Name and mailing address of the ISA European Patent Office, P.O. Box 5018 Patenthaus 2 NL - 2200 MB Rijswijk Tel. (+31-70) 540-5000, Tlx. 31 851 000 01 Fax: (+31-70) 540-5016		Authorized officer Wiltink, J

Form PCT/ISA210 (second sheet) July 1993

INTERNATIONAL SEARCH REPORT

Int. national Application No
PCT/US 99/04134

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	GB 2 216 307 A (ARDENT COMPUTER CORP) 4 October 1989 (1989-10-04)	13, 14, 17, 18, 21-23, 26, 27, 30
A	abstract page 3	1-12, 15, 16, 19, 20, 24, 25, 28, 29
A	US 4 434 461 A (PUHL LARRY C) 28 February 1984 (1984-02-28) abstract column 2, line 3 - line 45 column 6, line 19 - line 40 claim 1	1-30
A	MILUTINOVIC V ET AL: "Impacts of GaAs on microprocessor architecture" PROCEEDINGS OF IEEE INTERNATIONAL CONFERENCE ON COMPUTER DESIGN: VLSI IN COMPUTERS. ICCD '85 (CAT. NO.85CH2223-6), PORT CHESTER, NY, USA, 7 - 10 October 1985, pages 30-40, XP002110923 1985, Washington, DC, USA, IEEE Comput. Soc. Press, USAISBN: 0-8186-0642-8 page 34, right-hand column, line 27 - page 36, left-hand column, last last	1-12
A	US 4 694 433 A (WIEDMANN SIEGFRIED K) 15 September 1987 (1987-09-15) abstract column 2, line 42 - column 3, line 19	1-30

From PCT/BA216 (continuation of second sheet) (July 1999)

INTERNATIONAL SEARCH REPORT

Information on patent family members

Int. Appl. No.

PCT/US 99/04134

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 4410939 A	18-10-1983	JP 56016248 A	17-02-1981
GB 2216307 A	04-10-1989	DE 3906327 A	14-09-1989
		FR 2628237 A	08-09-1989
		JP 2010467 A	16-01-1990
US 4434461 A	28-02-1984	NONE	
US 4694433 A	15-09-1987	EP 0162934 A	04-12-1985
		AT 47928 T	15-11-1989
		CA 1232354 A	02-02-1988
		JP 60246090 A	05-12-1985

フロントページの続き

(71)出願人 1730 North First Street, San Jose, CA, USA

(72)発明者 アルフレート エーダー
ドイツ連邦共和国 フリートベルト グリ
ューンテンシュトラッセ 3

Fターム(参考) 5B033 AA01 AA15 BE00 DD08 EA00
5B098 BB01 BB18 DD01 DD08 GA04
GD02